



DECUS

PROGRAM LIBRARY

DECUS NO.	8-249
TITLE	OSCILLOSCOPE VECTOR GENERATOR
AUTHOR	J. H. Boardman
COMPANY	South Dakota School of Mines & Technology Rapid City, South Dakota
DATE	August 1, 1969
SOURCE LANGUAGE	PAL-D

20380
DISTRIBUTION

1b

101

20

1

OSCILLOSCOPE VECTOR GENERATOR

DECUS Program Library Write-up

DECUS No. 8-249

ABSTRACT

VCT is a subroutine requiring exactly one page (128 words) of memory. Its purpose is to draw a linear vector from any point on an oscilloscope display to any other point. There are 5 arguments in the subroutine calling sequence which specify the starting point (X_1, Y_1), the ending point (X_2, Y_2) and a stepsize. The stepsize (range 1 to 7) determines the point density along the vector. Minimum hardware requirements are EAE and Type 34D display.

DESCRIPTION

VCT is a subroutine which will draw a line between any two points on the Type 34D oscilloscope. A line will be drawn between the points (X_1, Y_1) and (X_2, Y_2) where X_1, Y_1, X_2, Y_2 are integers within the range 0 to 1023 (decimal). The calling sequence for VCT is:

```
JMS VCT
      X1
      Y1
      X2
      Y2
      STEPSIZE
```

The stepsize determines the distance between intensified points. The stepsize must be greater than 0 and less than 8. Execution time of VCT is determined largely by the length of the line and the stepsize. The resolution of the Type 34D is such that using a stepsize of 4, VCT generates a line that appears to be solid. A stepsize of 8 will generate a line in which individual intensified points will be visible.

THEORY OF OPERATION

The equation for a straight line is:

$$Y = MX + B$$

where: M = slope of the line
 B = Y axis intercept

Given two points (X_1, Y_1) and (X_2, Y_2), the slope is found by:

$$(1) \quad M = \frac{(Y_2 - Y_1)}{(X_2 - X_1)}$$

and the intercept is:

$$(2) \quad B = Y_1 - MX_1$$

VCT uses (1) and (2) to set up constants which can be used to determine all intermediate points along the specified line vector.

A difficulty arises however, when M is very large (vector approaching the vertical). In this situation, the result of the multiplication MX may exceed the 12 bit capacity of the PDP-8 word. To avoid the time consuming manipulation of double precision calculations, a trick has been used so that the value of M is always less than or equal to 1. The trick is simply that whenever M is greater than 1, X and Y values are interchanged and the basic equation:

$$X = MY + B$$

is used instead. To make the line plot correctly however, the DXL and DYL instructions must also be interchanged.

When the constants M and B have finally been determined, the display loop is entered. The first intensified point is (X_1, Y_1) and thereafter the stepsize is added to the X_1 value, the basic equation is solved for the corresponding Y value, and the new point is intensified. This process continues until the X value of the last intensified point is within one stepsize of X_2 .

Each execution of the display loop required approximately 148 microseconds. A maximum length vector with minimum stepsize would require about .16 seconds to be displayed. The resolution of the Type 34D is such that a stepsize of 4 will still yield a vector which appears to be a solid line, but the display time for the example above would be reduced by a factor of 4 to about .04 seconds.

RESTRICTIONS

Note that this routine requires an EAE (Type 182, or KE8/I) and, of course, the Type 34D (or VC8/I) oscilloscope control.

PROGRAMMING LANGUAGE

The routine was originally written with the PALD disk monitor system assembler in mind. However, the only incompatible statement in the subroutine for the PAL III assembler is the PAGE pseudooperator which occurs in the first line of text. The VCT origin should, however, occur at the beginning of a computer page. VCT is exactly one page (128 words) long.

The mnemonics MQL, MUY, DVI, MQA, DXL, DYL, and DIX are used in the subroutine and must be user defined in the fixed symbol table of the assembler in use.

/ROUTINE TO GENERATE LINE BETWEEN (X1,Y1) AND (X2,Y2)
 /ON THE DISPLAY UNIT
 /AUGUST 5, 1969
 /

VCT, 00
 CLA

/INITIALIZE DISPLAY COMMANDS

TAD VCTIMP+7	/GET A DXL
DCA VCTDXY	
TAD VCTIMP+6	/GET A DYL
DCA VCTDXY+5	
TAD VCTIMP+5	/-5
DCA VCTIMP	/USE AS COUNTER
TAD VCTIMP+4	/GET ADR(VCTX1)
DCA VCTMUY	/USE AS POINTER
TAD I VCT	/PICK UP ARGUMENTS
ISZ VCT	
AND VCTIMP+3	/MASK WITH 1777
DCA I VCTMUY	
ISZ VCTMUY	
ISZ VCTIMP	
JMP .-6	

/ELIMINATE ZERO STEP SIZE

TAD VCTX1+4
 SNA
 IAC
 DCA VCTX1+4

/CALCULATE SLOPE OF LINE

JMS VCTGDX	
SNA	/AC=ABS(X2-X1)
IAC	/DISALLOW X1=X2
DCA VCTDX	
RAR	/GET LINK
DCA VCTMUY+6	/=4000 IF (X2-X1) IS +VE
TAD VCTX1+1	/CALC ABS(Y2-Y1)
CIA CLL	
TAD VCTX1+3	
SPA	
CIA	
SNA	
IAC	/DISALLOW Y2=Y1
DCA VCTDY	
TAD VCTMUY+6	
SNA CLA	/SLOPE ALREADY -VE?
CML	/REVERSE SKIP SENSING IF SLOPE ALREADY -VE
SNL	/NO - SET SLOPE
TAD VCTIMP+2	/41+NOP=CIA (-VE SLOPE)
TAD VCTIMP+1	/NOP
DCA VCTMUY+6	
TAD VCTDY	
MQL DVI	/CALC ACTUAL SLOPE = DY/DX

```

VCTDX, 00
        DCA VCTMUY+2      /NUMERATOR
        TAD VCTDX
        DCA VCTMUY+4      /DENOMINATOR
        MQA
        SNA CLA
        JMP VCTA

```

/SLOPE .GE. 1 - REVERSE DYL, DXL INSTS, RECALC SLOPE

```

VCTDY, TAD VCTDX
        MQL DVI           /SET SLOPE = DX/DY
        00
        SNA
        TAD VCTDY        /SLOPE=1
        DCA VCTMUY+2      /NUMERATOR
        TAD VCTDY
        DCA VCTMUY+4      /DENOMINATOR
        TAD VCTX1+1      /INTERCHANGE X1,Y1
        DCA VCTTMP
        TAD VCTX1
        DCA VCTX1+1
        TAD VCTTMP
        DCA VCTX1
        TAD VCTX1+3      /PUT Y2 INTO X2
        DCA VCTX1+2

```

```

        TAD VCTTMP+6      /GET A DYL
        DCA VCTDXY
        TAD VCTTMP+7      /GET A DXL
        DCA VCTDXY+5

```

```

VCTA,  JMS VCTGDX
        CLA
        TAD VCTX1+4
        CIA
        DCA VCTTMP-2      /END POINT TEST VALUE
        TAD VCTX1+4      /GET STEP SIZE
        SNL
        CIA               /SET TO STEP X BKWDS (X2<X1)
        DCA VCTX1+4

```

/CALCULATE INTERCEPT

```

        TAD VCTX1
        JMS VCTMUY
        CIA               /B=YI-M*X1
        TAD VCTX1+1
        DCA VCTTMP        /INTERCEPT

```

/BEGIN DISPLAY

```

VCTGO, JMS VCTGDX
        TAD VCTTMP-2
        SPA CLA
        JMP I VCT         /EXIT WHEN ABS(X2-X) < FE STEPSIZE
        TAD VCTX1
        TAD VCTX1+4

```



```

VCTDXY,  JMP .           /DXL OR DYL
          DCA VCTX1
          TAD VCTX1       /X
          JMS VCTMUY      /M*X
          TAD VCTTMP      /M*X+B
          JMP .           /DYL OR DXL
          DCA VCTX1+1
          DIX             /INTENSIFY
          JMP VCTGO

```

/ROUTINE TO CALC ABS(X2-X1)

```

VCTGDX,  00
          TAD VCTX1
          CIA CLL
          TAD VCTX1+2
          SPA
          CIA
          JMP I VCTGDX

```

/ROUTINE TO MULT AC BY SLOPE

```

VCTMUY,  00
          MQL MUY
          00             /NUMERATOR
          DVI
          00             /DENOMINATOR
          CLA MQA
          JMP .           /NOP OR CIA (+VE OR -VE SLOPE)
          JMP I VCTMUY

```

```

VCTX1,   00
          00
          00
          00
          00

```

```

VCTTMP,  00

```

/VCT CONSTANTS

```

NOP
41
1777
VCTX1
-5
DYL
DXL

```

